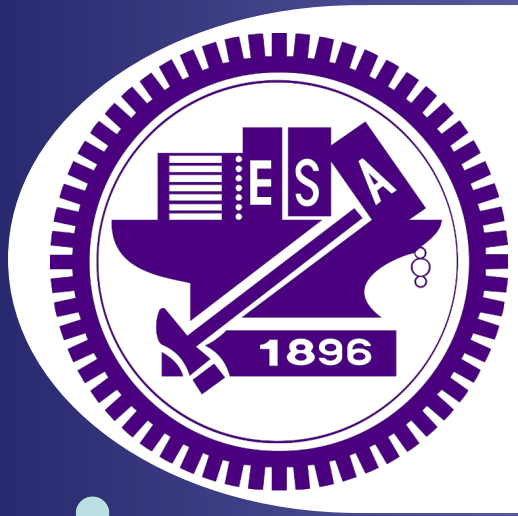


Visual Recognition using Deep Learning
2025 Spring, Homework 2

Release Date: 2025/03/26 12:00

Homework 2

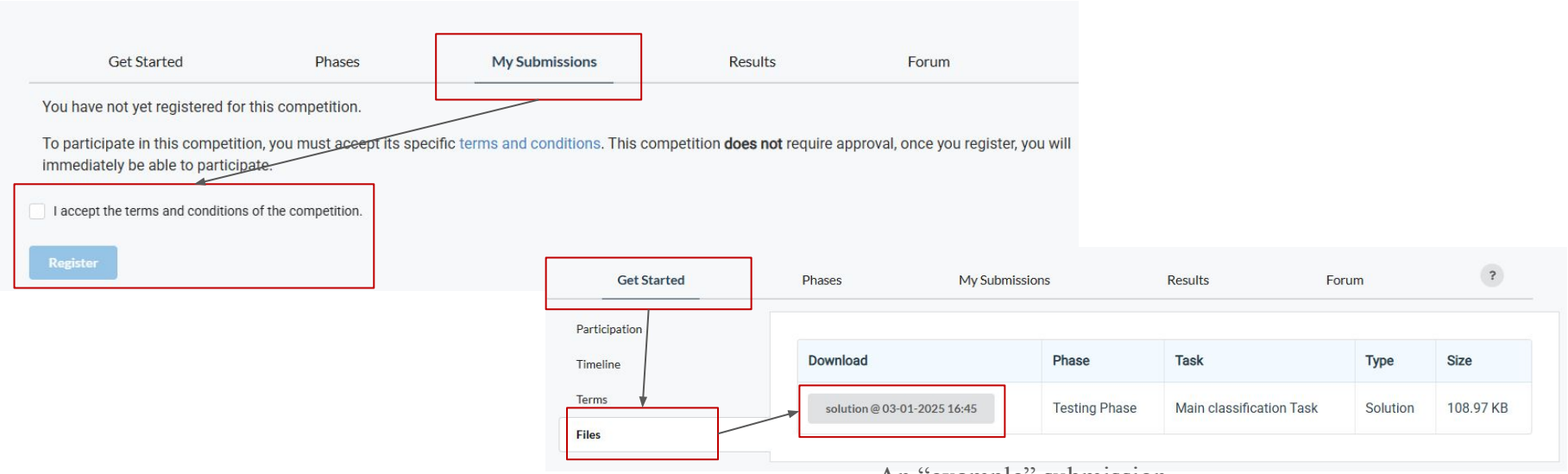
- Deadline: 23:59, 04/16 (Wed), 2025
- **Participate the competition** (80%): Digit Recognition Problem
  - Participant the competition on the CodaBench and get the highest score as possible. (70%)
  - Code reliability & quality (10%)
- **Report and code** (20%): Document your method and findings.
  - Report
    - In PDF format and written in English. (5pt penalty)
    - Introduction to your method (e.g., data pre-processing, model architecture, hyper-parameters)
    - Innovative ideas or additional experiments to further improve the model.
  - Code
    - Zip your code (.py) alone with report - Submit to E3.
    - You should also put your code on your GitHub repository and provide the link in the report.

Links

- Link to the dataset
- Link to the competition

How to participate the competition and do submission

1. Register an account on CodaBench
   a. When registering the account, please use your studentID as the UserName
2. After you click the competition link, go to My Submissions, and join the competition

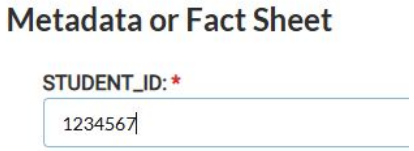


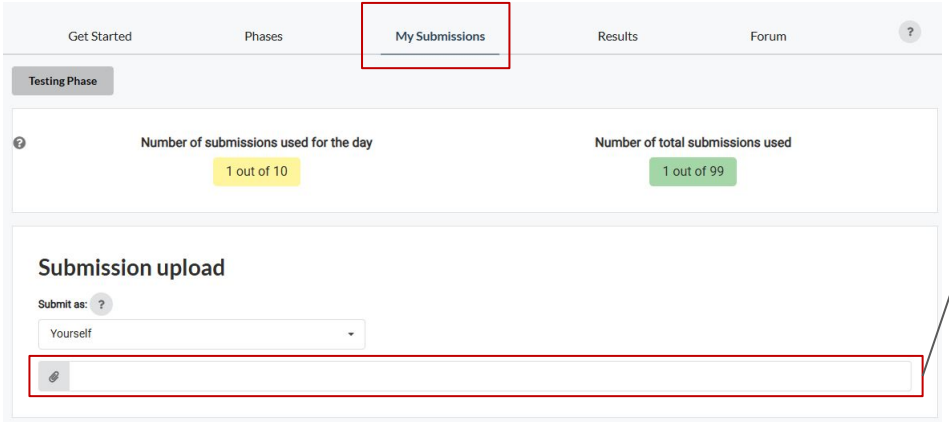
An “example” submission

How to participate the competition and do submission

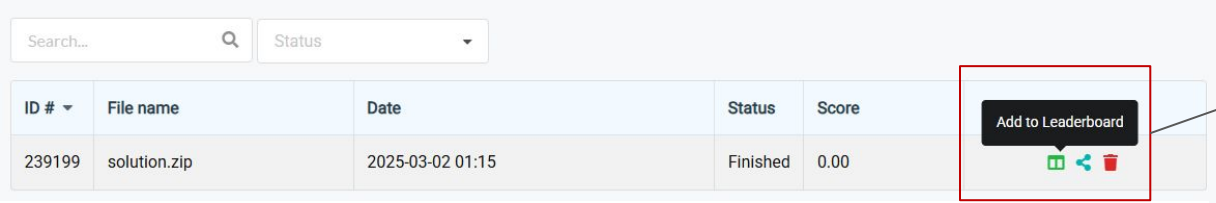
3. Submit your results and don't forget to "Add to Leaderboard"

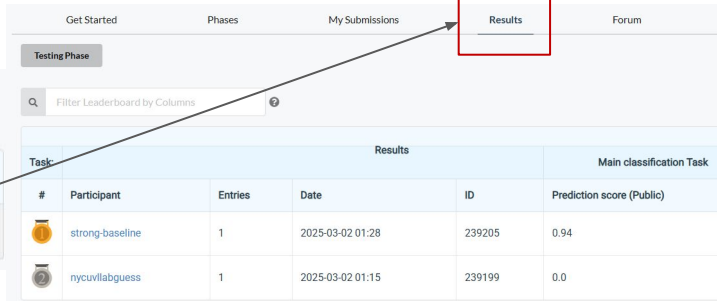
4. Don't forget to check your results can be found on the leaderboard





No limitation for the name of the "zip" file; however, inside the zip file, your **result files must be named "pred.json" and "pred.csv"**





Coding Environment

- Recommnedation: Python 3.9 or higher
- Tips
  - We recommend you to use **virtual environments** when implementing your homework assignments.
  - Here are some popular virtual environment management tools
    - Poetry
    - Conda
    - Virtualenv

Numpy & PyTorch

- Numpy Tutorial: Link
- PyTorch Tutorial: Link
    - Free to use any modules and functions

Task and Dataset

- Task: Digit Recognition
- Dataset:
  - RGB images
    - Training / Validation: 30,062 / 3,340
    - Test: 13,068
- Target:
  - task1: The class and bounding box of each digit in the image, e.g., the bboxes of “4” and “9”.
  - task2: The number of detected digits in the image, e.g., the number “49”.
- **Additional Requirements / Limitations** **(15 pts penalty, each)**
  - No external data (i.e., data from other sources) allowed.
  - You can **only** use Faster R-CNN as the model in this task. Faster R-CNN consists of a backbone, a neck (Region Proposal Network) and a head. Modification to each part is allowed, and elaborating your modification in the report may help you get good report score.
  - Not a requirement: pretrained weights is allowed.

Task and Dataset

- Use Faster R-CNN to recognize digits in the image.
  - The training and validation labels are JSON files in COCO format.
  - The bounding boxes are described in the format [x_min, y_min, w, h] without normalization. **The category id starts from 1.**
  - **Task1:** Detect each digit in the image. The submission file should be a JSON file in COCO format named **pred.json**. Specifically, it should be a list of labels, where each label is represented as a dictionary with the keys: image_id, bbox, score, and category_id.
  - **Task2:** Recognize the entire digit in the image. The submission file, **pred.csv**, should contain two columns: image_id and pred_label.

2.png

Task 2

image_id	pred_label
1	-1
2	112
3	9

Task 1


```
{
    "image_id": 2,
    "bbox": [
        98.08123016357422,
        14.60795783996582,
        10.317886352539062,
        20.47014808654785
    ],
    "score": 0.9570107460021973,
    "category_id": 3
},
{
    "image_id": 2,
    "bbox": [
        54.659828186035156,
        7.852010726928711,
        12.288619995117188,
        24.77677345275879
    ],
    "score": 0.7752974629402161,
    "category_id": 2
},
{
    "image_id": 2,
    "bbox": [
        90.57273864746094,
        15.257061958312988,
        9.276664733886719,
        19.51452922821045
    ],
    "score": 0.6907596588134766,
    "category_id": 2
},
```

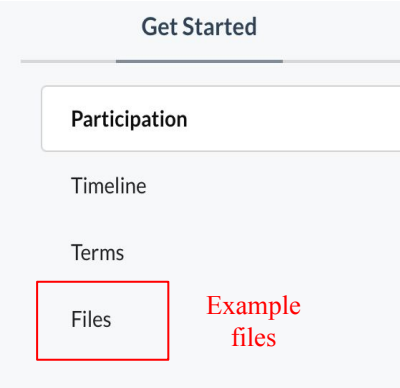


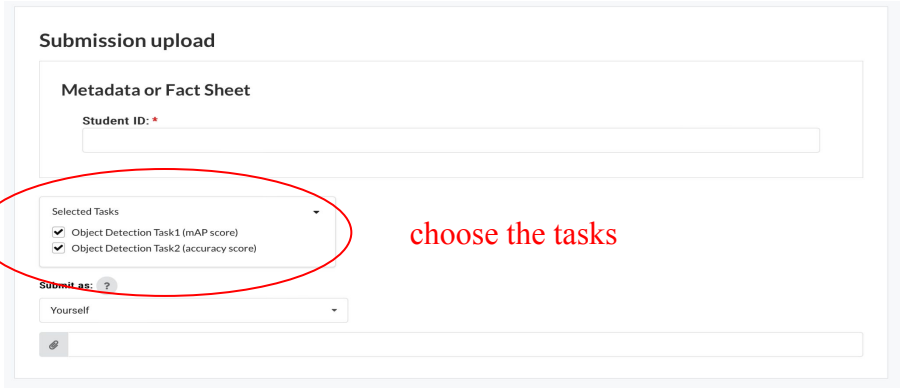
```
"categories": [
    {
        "id": 1,
        "name": "0"
    },
    {
        "id": 2,
        "name": "1"
    },
    {
        "id": 3,
        "name": "2"
    },
    {
        "id": 4,
        "name": "3"
    },
    {
        "id": 5,
        "name": "4"
    },
    {
        "id": 6,
        "name": "5"
    },
    {
        "id": 7,
        "name": "6"
    },
    {
        "id": 8,
        "name": "7"
    },
    {
        "id": 9,
        "name": "8"
    },
    {
        "id": 10,
        "name": "9"
    }
]
```

Remarks

- In Task 1, the bounding boxes are described in the format [x_min, y_min, w, h] without normalization. **Remember the difference between numbers and their category_id.**
- In Task 2, you can only use the results from Task 1 **(pred.json)** as input to predict the whole number. If your model predict no number in an image, you should write -1 in the pred_label. Otherwise, you will recieve n/a.

image_id	pred_label
1	-1
2	112
3	9





Grading Policy - Report (20%)

- Format: PDF, written in English. (-5pts if not followed)
- Sections that you should include
  - **Introduction** to the task and core idea of your method
  - **Method:** Descbible how you pre-process the data; what is your model architecture, and hyperparameters, etc.
    - You need to describe the architecture used in the backbone, neck (RPN), and head of Faster R-CNN.
  - **Results**: your findings / model performance (e.g., training curve, confusion matrix, etc.)
  - **References**: Your method references (paper / Github sources, must include if you use any.)

15pts

We encourage you to stand on the shoulders of giants - only clone and run it is not enough.

5pts

- **Among various architectures, why do you choose this one as your module? What are its pros and cons?**
- **Additional experiments** to explore better performance
  - Simply tuning the hyper-parameters doesn't count (e.g., batch-size, LR, different optimizers)
  - Hint: Try to add/remove some layers, use different design, use different loss functions, etc.
- You should 1) include your hypothesis (why you do this), 2) How this may (or may not) work, and 3) The experiment results and their implications.

Python Coding Style Guide Reference

1. PEP8
2. Google Python Style

Grading Policy - Code Reliability (10%)

1. Please follow the PEP8 instructions and lint your code.
2. Push your code to the GitHub
   - It should contains a README.md to introduce this work (And your StudentID).

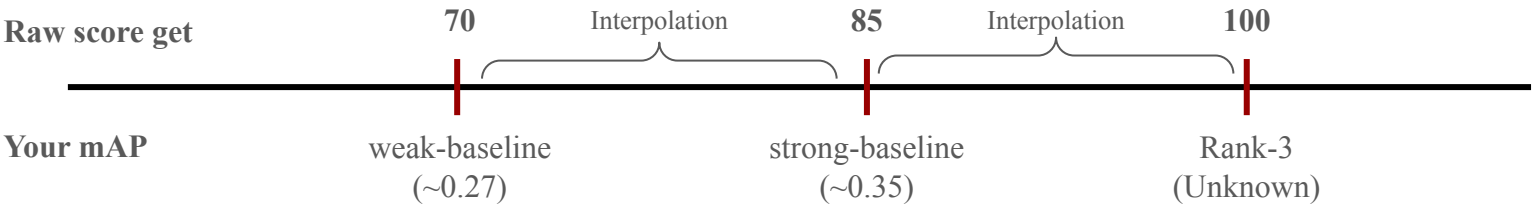
An example: README.md

Grading Policy - Task 1 (35%)

We will use **private (hidden) leaderboard** to evaluate the performance (the distribution is similar for data in public and private set.) The public leaderboard is for you as reference.

Your score (competinion):

- Less than weak-baseline (mAP < w.baseline): S = 0
- Between weak-baseline and strong baseline (mAP >= w.baseline & mAP < s.baseline): (70 + (X - w.baseline) / (s.baseline - w.baseline) * (85 - 70)) * 0.35
- Between strong-baseline and Rank3: (85 + (X - s.baseline) / (mAP.rank3 - s.baseline) * (100 - 85)) * 0.35
- Rank1,2,3 = 100 * 0.35

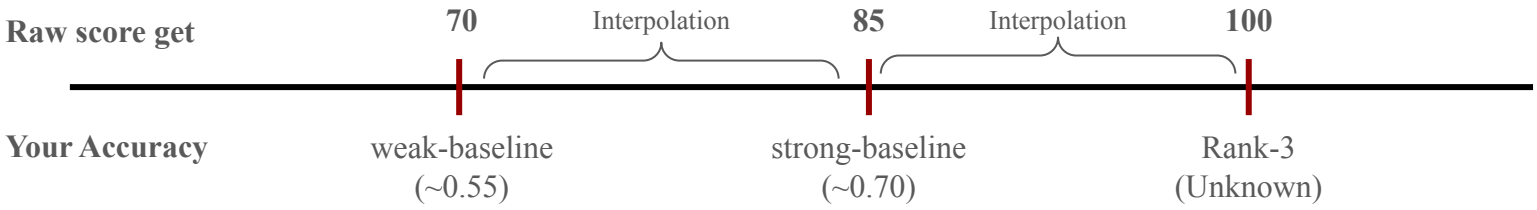


Grading Policy - Task 2 (35%)

We will use **private (hidden) leaderboard** to evaluate the performance (the distribution is similar for data in public and private set.) The public leaderboard is for you as reference.

Your score (competinion):

- Less than weak-baseline (Acc < w.baseline): S = 0
- Between weak-baseline and strong baseline (Acc >= w.baseline & Acc < s.baseline): (70 + (X - w.baseline) / (s.baseline - w.baseline) * (85 - 70)) * 0.35
- Between strong-baseline and Rank3: (85 + (X - s.baseline) / (Acc.rank3 - s.baseline) * (100 - 85)) * 0.35
- Rank1,2,3 = 100 * 0.35



Submission

- Compress your **code** and **report** into a **.zip file** and submit it to E3.
  - Don’t forget to push your code to GitHub. And your GitHub link should be written in the report.
- Report should be written in English.
- <STUDENT ID>_HW2.zip
  - codes (.py, folders, etc)
  - <STUDENT ID>_HW2.**pdf** (NO .doc, .docx or others format)

- Don’t put the data (e.g. x.jpg / train.csv / test.csv) and model checkpoints into submission file (-5 if not followed)

Other rules

- **Late Policy**: A penalty of **20 points** per additional late day. (-20pt / delayed.day)
  - For example, If you get 90 points but delay for two days, your will get only 50 points!

- **No Plagiarism**: You should complete the assignment by yourself. Students engaged in plagiarism will be penalized heavily. Super serious penalty.
  - e.g. -100pt for the assignment or failed this course, etc
  - Report to academic integrity office

FAQs

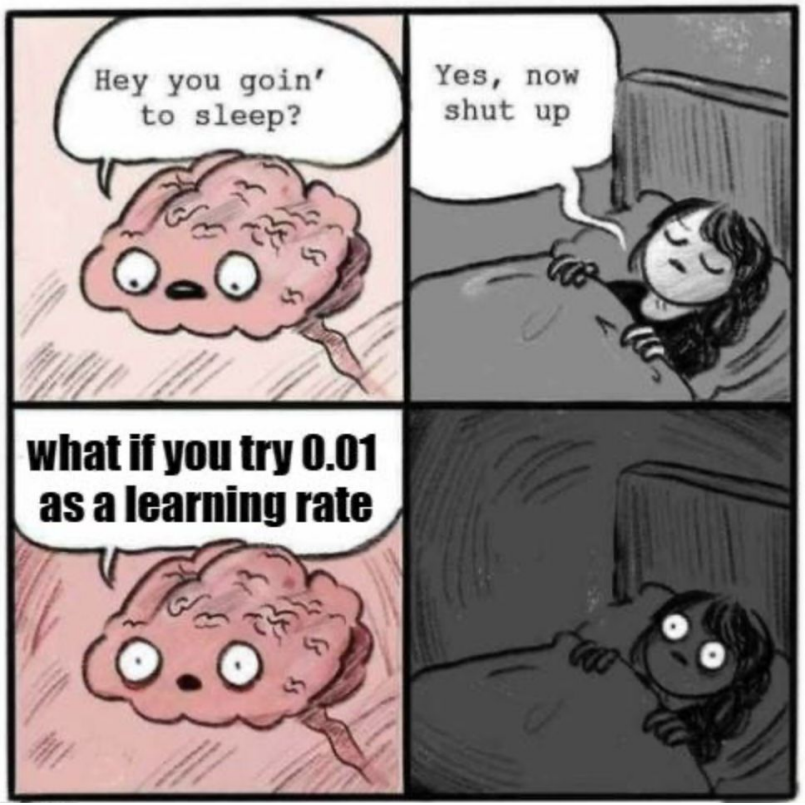
- Can I use any library/package/framework from GitHub or other resources?
  - **Yes**, we encourage you to learn how to leverage existing knowledge on your own task
    - e.g., Github of published works and model zoo from Torchvision
    - Focus on how to step forward from them - That's why part of scores comes from your competition ranks
  - You **should not copy-and-paste from your classmates** (Plagiarism)
- How to handle the GPU Out-of-Memory (OOM) issue?
  - Easy answer - Make your batch size smaller or make your model smaller.
  - Advanced methods: Try to figure it out by yourself. (Many online resources and AI-assistance)

FAQs

- If I don’t have my own GPU – Use Google Colab
  - It should be 12 hours, please check this discussion in the stackoverflow
  - And some tricks here may make it longer.

- If you have other questions, ask on **E3 forum** first! We will reply as soon as possible.

It’s your turn! Have Fun!



Team-up for the final project!

3/26 (Wed) - 4/23 (Wed) – After 4/24, we will random assign

Find 4 classmates to team up. [Link to the form]

- “Team Member 1” will be the leader (We’ll contact leader when needed.)
- Feel free to invite/join using E3 discussion board. (Just use homework discussion board)

Report order may be related to “topic” and in a random order - announce after the topic is selected.

A	B	C	D	E	F	G
GroupID	GroupName	TeamMember1	TeamMember2	TeamMember3	TeamMember4	Selected Topic
Group1						
Group2						
Group3						
Group4						
Group5						
Group6						
Group7						
Group8						
Group9						
Group10						

Group Name
Can be either Chinese English
No affect to the grading

[Student ID], [Your Name]

Topic will be released in the future.